AMENDMENTS TO THE CLAIMS

Please amend the claims as follows.

1. (Currently Amended) A computer-implemented method for a system having distributed collaborating components, comprising:

restricting direct interaction between a first distributed collaborating component[[s]] and a second distributed collaborating component by introducing an application-independent interface between the first and second distributed collaborating components; and

invoking a service from the application-independent interface in order to enable interaction between the first and second distributed collaborating components; and wherein invoking a service from the application-independent interface comprises

sending a usage specification as an argument to the application-independent interface, wherein the usage specification comprises a server object and a plurality of attributes associated with the server object that are requested by the first distributed collaborating component from the second distributed collaborating component,

wherein the application-independent interface is configured to:

interpret the usage specification to determine the plurality of attributes to fetch from the second distributed collaborating component;

obtain the plurality of attributes from the second distributed collaborating component; and

provide the first distributed collaborating component with the plurality of attributes.

2. (Canceled)

3. (Previously Presented) The method of claim 1, wherein the application-independent interface has a capability to interpret the usage specification at runtime.

4. (Currently Amended) A computer-implemented method for a system having distributed collaborating components, comprising:

restricting direct interaction between a first distributed collaborating component[[s]] and a second distributed collaborating component by introducing an application-independent interface between the first and second distributed collaborating components; and

invoking a service from the application-independent interface in order to enable interaction between the first and second distributed collaborating components; and

sending a usage specification as an argument to the application-independent interface, wherein the usage specification comprises a server object and a plurality of attributes associated with the server object that are requested by the first distributed collaborating component from the second distributed collaborating component wherein invoking a service from the application-independent interface comprises sending a logic execution specification as an argument to the application-independent interface, wherein the application-independent interface is configured to:

interpret the usage specification to determine the plurality of attributes to fetch from the second distributed collaborating component;

obtain the plurality of attributes from the second distributed collaborating component; and

provide the first distributed collaborating component with the plurality of attributes.

5. (Original) The method of claim 4, wherein the application-independent interface has a capability to interpret the logic execution specification at runtime.

6. (Currently Amended) A method for a distributed system having a client and a server, comprising:

interposing a service layer between the client and the server, the service layer having a capability to interpret a usage specification sent as an argument from the client at runtime in order to enable interaction between the client and the server,

wherein the usage specification comprises a server object and a plurality of attributes associated with the server object that are requested by the client from the

server ~~the specification is one selected from the group consisting of a usage~~
~~specification and a logic execution specification~~; and
routing correspondence between the client and server through the service layer,
wherein the service layer is configured to:
interpret the usage specification to determine the plurality of attributes to fetch
from the server;
obtain the plurality of attributes from the server; and
provide the client with the plurality of attributes.

7. (Canceled)

8. (Currently Amended) The method of claim 7, further comprising:
the service layer fetching data from the server object based on the usage
specification.

9. (Currently Amended) The method of claim 8, wherein fetching data from the server
object comprises storing data fetched from the server in a proxy for the object.

10. (Currently Amended) The method of claim 7, further comprising the service layer
updating data in the server object based on the usage specification.

11. (Currently Amended) The method of claim 10, wherein updating data in the server
object comprises receiving data from the client and using data received from the client
to modify the attribute of the server object.

12. (Currently Amended) The method of claim 6, wherein the service layer is further
configured to interpret a logic execution specification comprising[[es]] logic for
invoking a method of ~~an object on~~ the server object.

13. (Currently Amended) The method of claim 12, wherein interpreting the logic
execution specification comprises invoking the method of the server object.

14. (Canceled)

15. (Canceled)

4

16. (Currently Amended) A computer-readable medium having recorded thereon instructions executable by a processor, the instructions for:

receiving a <u>usage</u> specification as an argument from a client component, <u>wherein the usage specification comprises a server object and a plurality of attributes associated with the server object that are requested by the client from the server;</u> ~~specification is one selected from the group consisting of a usage specification and a logic execution specification~~ and

interpreting the <u>usage</u> specification ~~in order~~ to enable interaction between the client component and a server component, wherein <u>interpreting</u> the <u>usage specification comprises:</u>

<u>determining the plurality of attributes to fetch from the server component;</u>

<u>obtaining the plurality of attributes from the server component; and</u>

<u>providing the client component with the plurality of attributes.</u>

17. (Canceled)

18. (Currently Amended) The computer-readable medium of claim 1<u>6</u>[[7]], further comprising:

instructions for fetching data from the <u>server</u> object based on the usage <u>specification.</u>

19. (Currently Amended) The computer-readable medium of claim 1<u>6</u>[[7]], further comprising:

instructions for updating data in the <u>server</u> object based on the usage <u>specification.</u>

20. (Previously Presented) The computer-readable medium of claim **16**, <u>further comprising instructions for</u> ~~wherein the~~ <u>receiving and interpreting a</u> logic execution specification compris<u>ing</u>[[es]] logic for invoking a method of an object on the server.

21. (Previously Presented) The computer-readable medium of claim **20**, wherein interpreting the logic execution specification comprises invoking the method of the object.

22. (Currently Amended) A distributed system, comprising:
a client component;

5

a server component having at least one object at runtime; and

a service layer between the client and the server component, the service layer having a
    capability to interpret a usage specification ~~of usage~~ comprising a plurality of
    attributes associated with ~~of~~ the at least one object, wherein the usage
    specification is sent as an argument at runtime, and

wherein interpreting the usage specification comprises:

    determining the plurality of attributes to fetch from the server component;

    obtaining the plurality of attributes from the server component; and

    providing the client component with the plurality of attributes.

23. (Currently Amended) The distributed system of claim **22**, wherein the service layer
    further has a capability of interpret~~ing~~ a logic execution specification ~~of logic
    executions~~ in the server component at runtime.

24. (Currently Amended) A distributed system, comprising:

a service means for providing application-independent services and for interpreting a
    usage specification and a logic execution specification, wherein the usage
    specification comprises a server object and a plurality of attributes associated
    with the server object that are requested by a client component from a server
    component;

[[a]] the client component that sends the usage specification and the [[a]] logic
    execution specification as an argument to the service means; and

[[a]] the server component that interacts with the service means in order to provide
    services to the client component,

wherein the service means is configured to:

    determine the plurality of attributes to fetch from the server component;

    obtain the plurality of attributes from the server component; and

    provide the client component with the plurality of attributes.

25. (Canceled)

26. (Canceled)